SDS 900891C
$1.00

DIAGNOSTIC PROGRAM MANUAL

SIGMA 5 AND 7 CPU DIAGNOSTIC SYSTEM (PATTERN)

PROGRAM NO. 704043C

October 1968

This publication supersedes SDS 900891B,
dated June 1968

SCIENTIFIC DATA SYSTEMS • 701 South Aviation Boulevard • El Segundo, Calif., 90245 • 213/772-4511

LIST OF EFFECTIVE PAGES

Total number of pages is 34, as follows:

Page No.	Issue	Page No.	Issue
Title	Original		
A	Original		
i thru ii	Original		
1-1 thru 1-2	Original		
2-1 thru 2-6	Original		
3-1 thru 3-4	Original		
4-1 thru 4-18	Original		

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

RELATED PUBLICATIONS

Publication Title	Publication No.
Sigma 5 Computer, Reference Manual	900959
Sigma 5 Computer, Technical Manual	901172
Sigma 7 Computer, Reference Manual	900950
Sigma 7 Computer, Technical Manual	901060
Sigma Symbol and Meta-Symbol, Reference Manual	900952

SECTION I
INTRODUCTION

1-1 SCOPE OF MANUAL

This manual describes the Pattern program of the CPU Diagnostic System designed for the Sigma 5 and Sigma 7 computers, manufactured by Scientific Data Systems.

This manual is made up of four sections. Section I is a general introduction to the Pattern program. Section II contains a discussion of operating procedures. Section III contains a detailed description of the program operation. Section IV contains the program's complete symbolic listing as generated by the Sigma Metasymbol Assembler.

1-2 PROGRAM OBJECTIVE

The purpose of the Pattern program is to detect and report any malfunctioning in any general register of each register page implemented, and to test the ability of the CPU to access and execute instructions from each core memory address.

1-3 GENERAL SPECIFICATIONS

Table 1-1 lists the general specifications for this program.

Table 1-2 shows the testing that must have been successfully completed before the Pattern program is run. Also listed are the testing prerequisites for the other Sigma 5/7 CPU diagnostic programs.

Table 1-1. General Specifications

Computer configuration	Any Sigma 5 or Sigma 7 computer with card reader or paper tape reader for program input
Memory size	4K minimum (4096 words)
Optional equipment	None

Table 1-2. Testing Prerequisites

Program	Prerequisite Program
Sense (Sigma 7 only)	None
Verify	None
Pattern	Verify
Auto	Verify, Pattern*
Suffix	Auto
Float	Auto
Interrupt	Auto
Memory Protect	Suffix
Map (Sigma 7 only)	Suffix

*For the Auto test to run, the block 0 register must be functioning correctly, as tested by the Pattern program

SECTION II
OPERATING INSTRUCTIONS

2-1 GENERAL

The Pattern diagnostic program consists of two general test sections: the register block test and the memory test. Each of these two sections in turn consists of two subsections. The first subsection of the register block test is to detect any error in any register of register blocks implemented (called type 1 errors). The second subsection is to report any errors (called type 2 errors) caused by interaction between the register blocks.

The memory test consists of the access test (type 3 errors) and the execution test (type 4 errors). The purpose is to reference all available memory locations as operand address and attempt to execute an instruction from each location.

2-2 LOADING PROCEDURE

Table 2-1 shows the control panel switch settings to be used for loading the program.

After the switches have been set as indicated in table 2-1, the following procedure is required:

a. Place the COMPUTE switch in the IDLE position.

b. Place the program card deck (or paper tape) in the appropriate reader and start the reader.

c. Clear memory by pressing the CPU RESET/CLEAR and SYS RESET/CLEAR buttons simultaneously.

d. Set the UNIT ADDRESS switches to the address of the input device.

Table 2-1. Switch Settings for Program Loading

Switch	Setting
CONTROL MODE	LOCAL
WATCHDOG TIMER	NORMAL
INTERLEAVE SELECT	NORMAL
PARITY ERROR MODE	CONT
AUDIO	ON
CLOCK MODE	CONT
ADDR STOP	Off
SENSE Switches	0

e. Press the LOAD switch.

f. Place the COMPUTE switch in the RUN position.

If the program is loaded with the switches set according to table 2-1, it will automatically branch to the starting location and begin running.

2-3 SUCCESS INDICATIONS

Provided that no errors occur, and if SENSE switches 1, 3, and 4 are all reset during the execution, the program will run continuously in cyclical fashion. A more positive determination of successful operation is obtained by setting SENSE switch 3, which brings the program to halt at a specified location, and then examining the pass count in general register 4.

See pages 2 and 3 of the program listing for a summary of SENSE switch indications and other displayed information.

2-4 ERROR INDICATIONS AND RECOVERY PROCEDURES

The program requires that general register 0 of page 0 is functioning properly. Any errors associated with this register will halt the program at error waits, the locations of which are specified on page 5 of the program listing under the heading of Additional Error Wait Locations.

The program cannot be continued further until register 0 is corrected. If replacing page 0 modules does not correct the malfunction, there could be interaction with either the REU (Register Extension Unit) or other internal pages.

Whenever program halt occurs due to a possible REU interaction, general register 0 will contain zeros in one or more bits of the interacting byte (or bytes). The operator is advised to pull out the BCR module in the CPU associated with failing byte. If the problem is still present, remove the modules for pages 1, 2, or 3 or the CPU. Sigma 5 may be affected by IOFM modules (see table 2-3 for all module locations).

If removing the BCR module does rectify the register 0 error, reinsert the module and follow the procedure given below to pinpoint the fault in the REU which causes the problem.

a. Ensure that the LT26 switches are set according to table 2-2 (LT26 module sets REU address).

b. If only some but not all bits in register 0 are failing, the most probable error is interference from page 4, 5, 6, or

7 in the REU at the faulty bits. Refer to table 2-3 for module locations. If, however, all bits of a byte (or bytes) of register 0 are in error, proceed according to the error analysis flow chart, figure 2-1, and perform the following test whenever called for after removing a given module.

> TEST: Check register 0 to see if the error is corrected. This is done by storing $FFFFFFFF_{16}$ in register 0, followed by a display of register 0. (It must be ensured that none of the display lamps are burned out.)
>
> If register 0 still shows the error, reinsert the removed module and continue with the procedure in the flow chart from the point the branch was made to this test.
>
> If, however, register 0 appears to be corrected, the removed module is a possible source of the error. Replace the module with a new one and examine register 0 again.
>
> If the error reappears after replacing the module, refer to Logic Equations (Dwg. No. 124817) or the simplified logic diagram, figure 2-2,for further troubleshooting.

Any error involving other registers in any page will terminate the program at error wait location 1E3. A complete set of error information as to the types and the source is available to the operator on page 4 of the program listing. Module locations are given in table 2-3.

Similarly, any core memory operand access errors or instruction execution errors will halt the program at the same wait location, 1E3. See page 5 of the listing for information associated with these errors.

Table 2-2. LT26 Switch Settings

Register Block Numbers	S3-2	S3-1	S2-2
4 through 7	0	0	1
8 through 11	0	1	0
12 through 15	0	1	1
16 through 19	1	0	0
20 through 23	1	0	1
24 through 27	1	1	0
28 through 31	1	1	1
Note: S2-1, S1-2, S1-1, S4-1, and S4-2 settings are irrelevant			

If the loop on current test option (see paragraph 2-6) is invoked before clearing the wait, the error halt will not occur during the loop.

2-5 OPTIONS

Options by means of SENSE switches are incorporated into the Pattern program to give the operator a more flexible tool for diagnosing failures.

2-6 LOOP ON CURRENT TEST

If SENSE switch 1 is set following an error, the test that detected a failure continues to repeat until SENSE switch 1 is reset.

2-7 REPEAT TEST WITH CURRENT PARAMETERS

If SENSE switch 2 is reset, the program calculates the number of implemented register pages and the core memory configuration on the first pass only and uses these values in the subsequent passes. If, however, SENSE switch 2 is set, the program recalculates on every pass. Switch 2 should thus be reset whenever the parameter change option is used (see paragraph 2-8).

2-8 HALT TO CHANGE REGISTER PAGE LIMITS OR MEMORY LIMITS

By setting SENSE switch 3, the operator halts the program at the end of the current program execution pass at location 1AB. At this time, the following information is displayed on the specified registers:

Register	Contents
0	Value of starting test page shown in bits 23-27
1	Last page implemented shown in bits 23-27
2	Minimum test core address
3	Maximum test core address
4	Pass count
5	Error count

Before proceeding, the contents of registers 0, 1, 2, or 3 may be altered to suit the operator's immediate need. The previous option (paragraph 2-7) is normally invoked in conjunction with this option.

2-9 NO HALT ON ERRORS

If SENSE switch 4 is set, no error wait will be recognized by the program except errors associated with register 0. The error count is maintained independent of this option.

2-10 PROGRAM REINITIALIZATION

With the COMPUTE switch at IDLE position, pressing the SYSTEM RESET/CLEAR button returns the program to the starting point. Reinitialization is also achieved by the execution of PCP interrupt.

Table 2-3. Register Block Module Locations

CPU								
Reg. Page No.	Sigma 5				Sigma 7			
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 0	Byte 1	Byte 2	Byte 3
0	09K	19K	05S	09T	01T	15T	01L	01Q
1	10K	18K	06S	10T	02T	16T	02L	02Q
2	11K	17K	07S	11T	03T	17T	03L	03Q
3	12K	16K	08S	12T	04T	18T	04L	04Q
BCR	13K	15K	09S	13T	05T	19T	05L	05Q
IOFM	05K	23K	01S	01T				
	06K	22K	02S	02T				
	07K	21K	03S	03T				
	08K	20K	04S	04T				
				05T				
				06T				
				07T				
				08T				

REU				
Register Page (Block) Number	Sigma 5/7			
	Byte 0	Byte 1	Byte 2	Byte 3
4, 8, 12, 16, 20, 24, 28	23A	19A	14A	04A
5, 9, 13, 17, 21, 25, 29	22A	18A	13A	03A
6, 10, 14, 18, 22, 26, 30	21A	17A	12A	02A*
7, 11, 15, 19, 23, 27, 31	20A	16A	11A	01A†

*Remove XT10 from 05A

†Remove XT10's from 05A, 09A

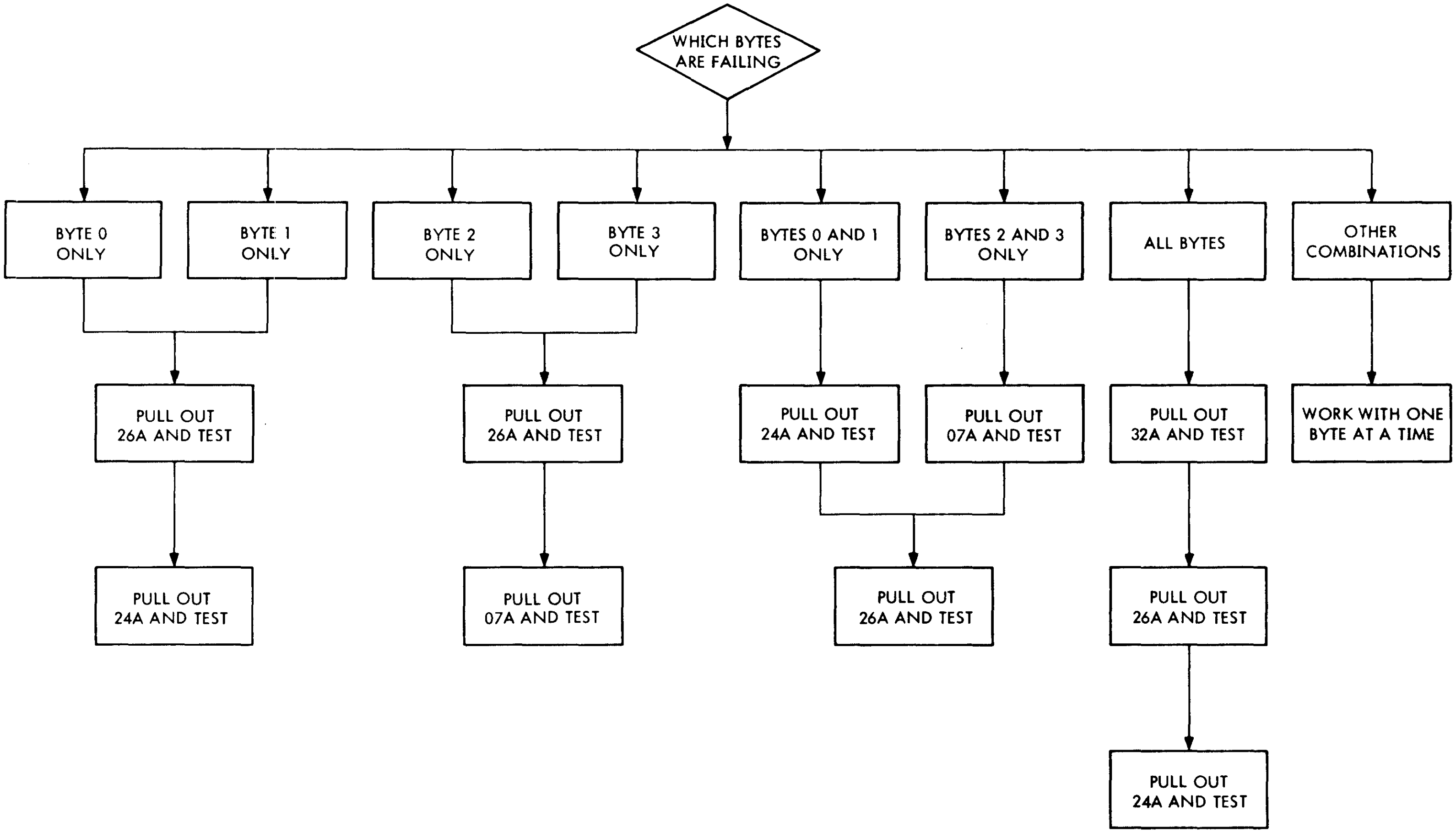

Figure 2-1. Error Analysis Flow Chart

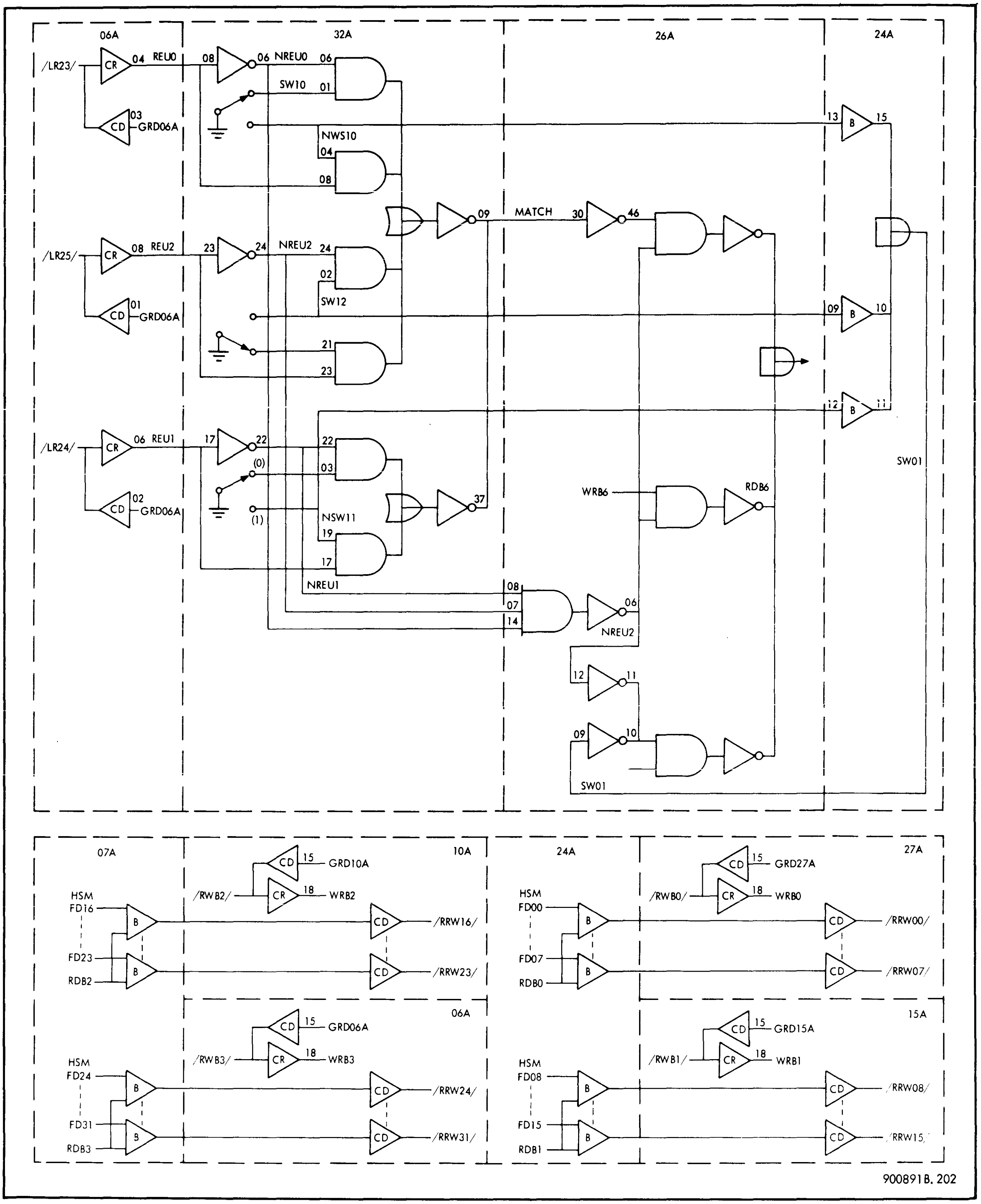

Figure 2-2. Simplified Logic Diagram

SECTION III
PROGRAM DESCRIPTION

3-1 GENERAL

After the program load, program execution begins with testing the proper operation of register 0 and then initialization and determination of the number of register pages implemented in the test computer. This is accomplished by loading a register in each page with X'555555', then examining the register for the content. A positive nonzero content indicates implementation. The value of the maximum implemented pages is stored for future reference.

The core memory size is determined by accessing each location until a nonexistent memory trap occurs at maximum address plus 1. The maximum address is also stored for future reference.

All above parameters can be altered by the operator (see paragraph 2-8).

The actual Pattern program tests are described in paragraphs 3-2 through 3-5.

3-2 INDIVIDUAL REGISTER TEST

Register 0 is independently tested at the beginning of the program. All other registers of all pages are then tested individually in turn. This is done by first storing a number pattern (called test pattern) in the register under test, followed by a routine which loads all remaining registers of the test page with the complement of the test pattern. The test register is then examined for containing the test pattern.

3-3 REGISTER PAGE ADDRESSING TEST

The purpose of this test is to detect errors arising when one page is affected by addressing another page. The test begins by loading all registers of all the implemented pages with ones. Then starting with the first page (test page), registers 0 and 8 of this page are cleared to zero. The register pointer is then advanced to scan succeeding pages one by one with registers 0 and 8 of each page tested for retaining the previously stored ones.

If no error is reported, the test page is incremented by one and the register pointer is set to the test page. Once again, registers 0 and 8 are cleared, followed by examination of succeeding pages. The process is repeated until all the implemented pages are tested.

3-4 MEMORY ACCESS TEST

Each memory location, from address 300_{16} to maximum, is set equal to its own address. All test memory is then sequentially read, with the content of each location compared against an independent counter that is incremented for each sequential memory access. Errors precipitate a halt at the common error wait.

3-5 MEMORY EXECUTION TEST

All of test memory is preloaded with an unconditional branch to an error routine. Starting with the lowest test address and proceeding sequentially through core to the maximum test address, each test location is loaded with a BIR instruction whose R field equals C and whose effective address is set to the normal program return. Register C is set to $7FFFFFFF_{16}$ and program control is transferred to the test location.

Successful execution of the BIR increments register C once and causes a branch back to the test program, where the process is repeated for the next memory location. Failure of the BIR to execute properly results in a branch to error; the program ultimately halts at the common error wait.

3-6 FLOW CHARTS

Figure 3-1 is a flow chart of the entire Pattern program.

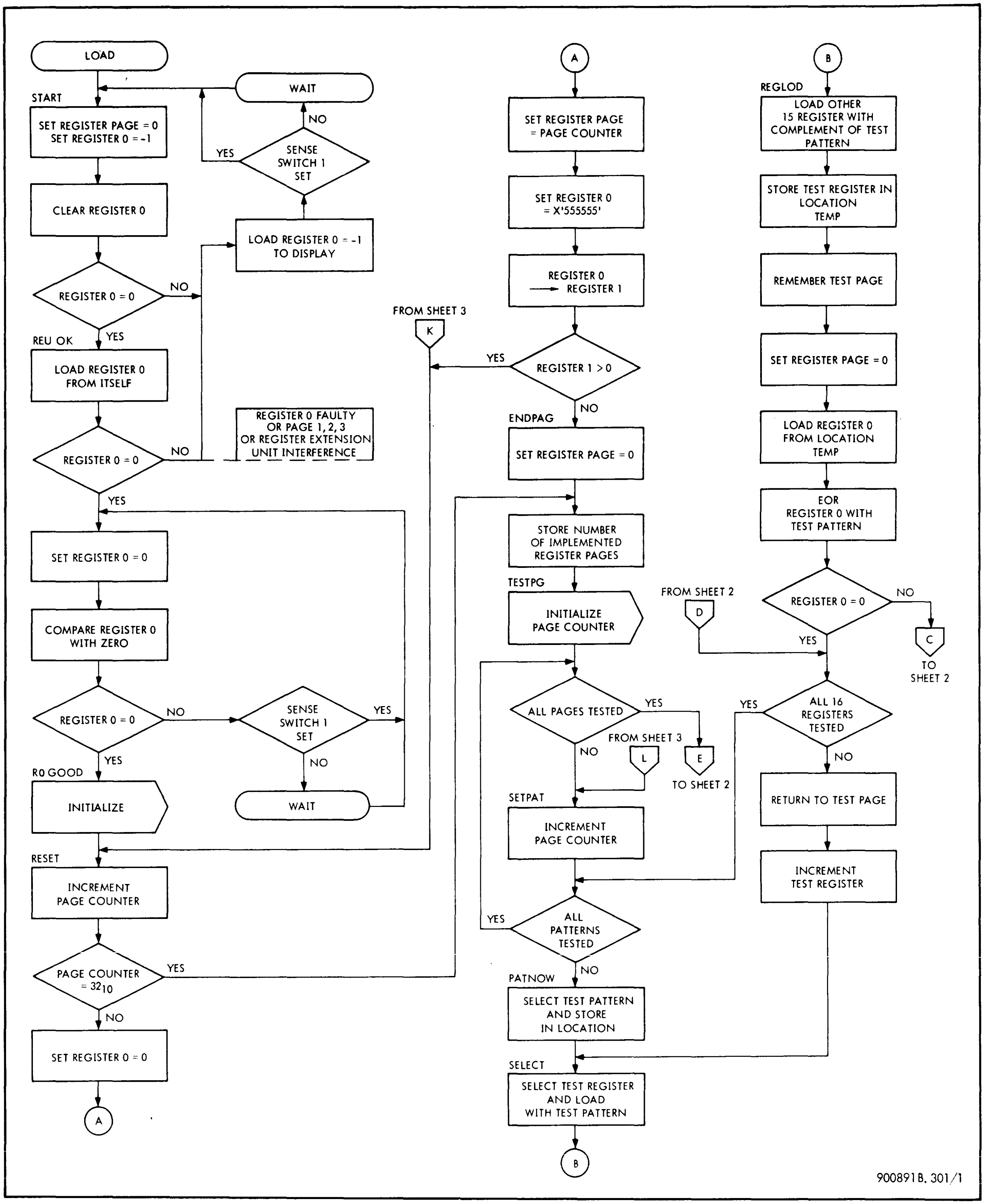

Figure 3-1. Pattern Program Flow Chart (Sheet 1 of 3)

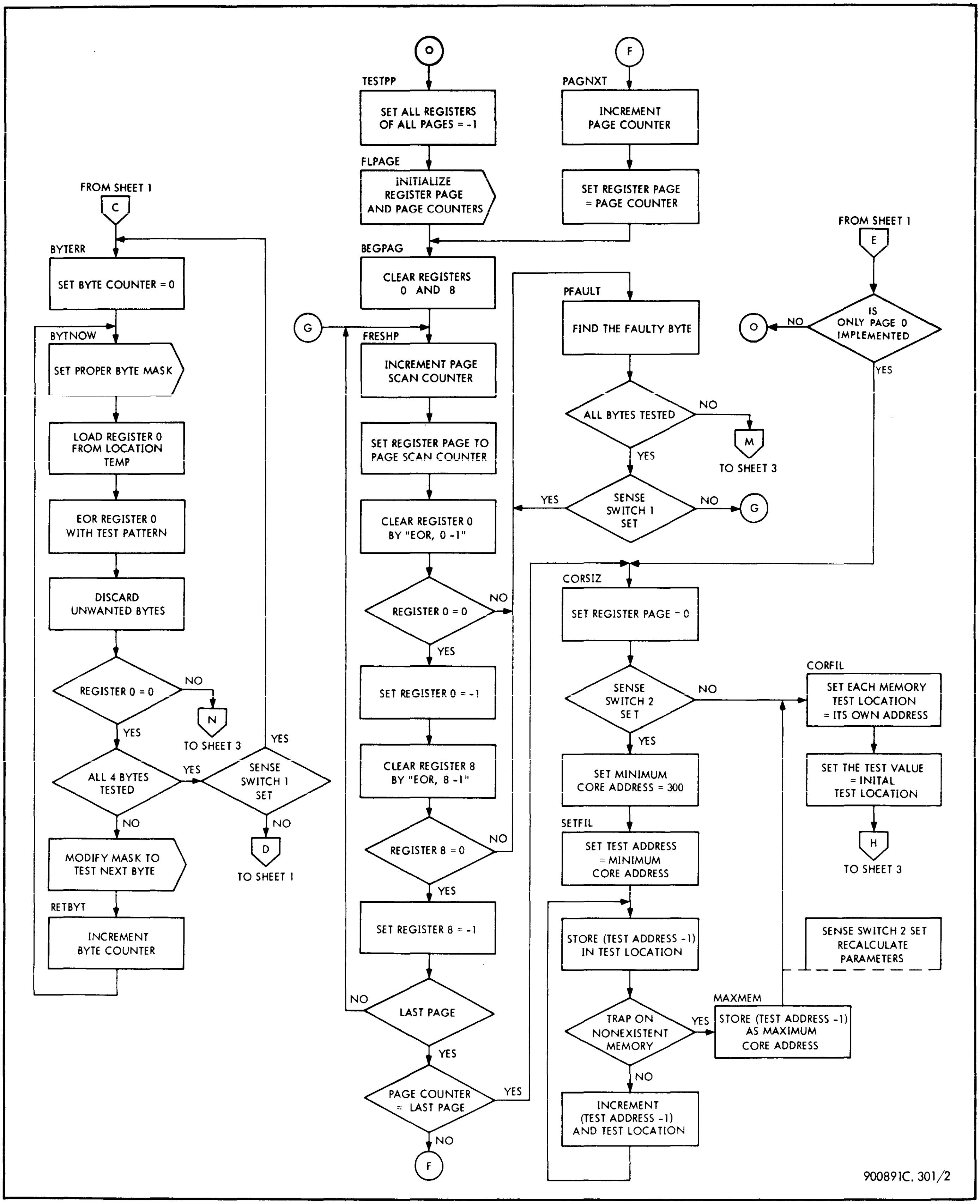
O
TESTPP
SET ALL REGISTERS OF ALL PAGES = -1
FLPAGE
INITIALIZE REGISTER PAGE AND PAGE COUNTERS
BEGPAG
CLEAR REGISTERS 0 AND 8
G
FRESHP
INCREMENT PAGE SCAN COUNTER
SET REGISTER PAGE TO PAGE SCAN COUNTER
CLEAR REGISTER 0 BY "EOR, 0 -1"
REGISTER 0 = 0
NO
YES
SET REGISTER 0 = -1
CLEAR REGISTER 8 BY "EOR, 8 -1"
REGISTER 8 = 0
NO
YES
SET REGISTER 8 = -1
LAST PAGE
NO
YES
PAGE COUNTER = LAST PAGE
YES
NO
F
F
PAGNXT
INCREMENT PAGE COUNTER
SET REGISTER PAGE = PAGE COUNTER
PFAULT
FIND THE FAULTY BYTE
ALL BYTES TESTED
NO
M
TO SHEET 3
YES
SENSE SWITCH 1 SET
YES
NO
G
CORSIZ
SET REGISTER PAGE = 0
SENSE SWITCH 2 SET
NO
YES
SET MINIMUM CORE ADDRESS = 300
SETFIL
SET TEST ADDRESS = MINIMUM CORE ADDRESS
STORE (TEST ADDRESS -1) IN TEST LOCATION
TRAP ON NONEXISTENT MEMORY
YES
NO
INCREMENT (TEST ADDRESS -1) AND TEST LOCATION
MAXMEM
STORE (TEST ADDRESS -1) AS MAXIMUM CORE ADDRESS
FROM SHEET 1
C
BYTERR
SET BYTE COUNTER = 0
BYTNOW
SET PROPER BYTE MASK
LOAD REGISTER 0 FROM LOCATION TEMP
EOR REGISTER 0 WITH TEST PATTERN
DISCARD UNWANTED BYTES
REGISTER 0 = 0
NO
N
TO SHEET 3
YES
ALL 4 BYTES TESTED
YES
SENSE SWITCH 1 SET
YES
NO
D
TO SHEET 1
NO
MODIFY MASK TO TEST NEXT BYTE
RETBYT
INCREMENT BYTE COUNTER
FROM SHEET 1
E
IS ONLY PAGE 0 IMPLEMENTED
NO
O
YES
CORFIL
SET EACH MEMORY TEST LOCATION = ITS OWN ADDRESS
SET THE TEST VALUE = INITAL TEST LOCATION
H
TO SHEET 3
SENSE SWITCH 2 SET RECALCULATE PARAMETERS
900891C. 301/2

Figure 3-1. Pattern Program Flow Chart (Sheet 2 of 3)

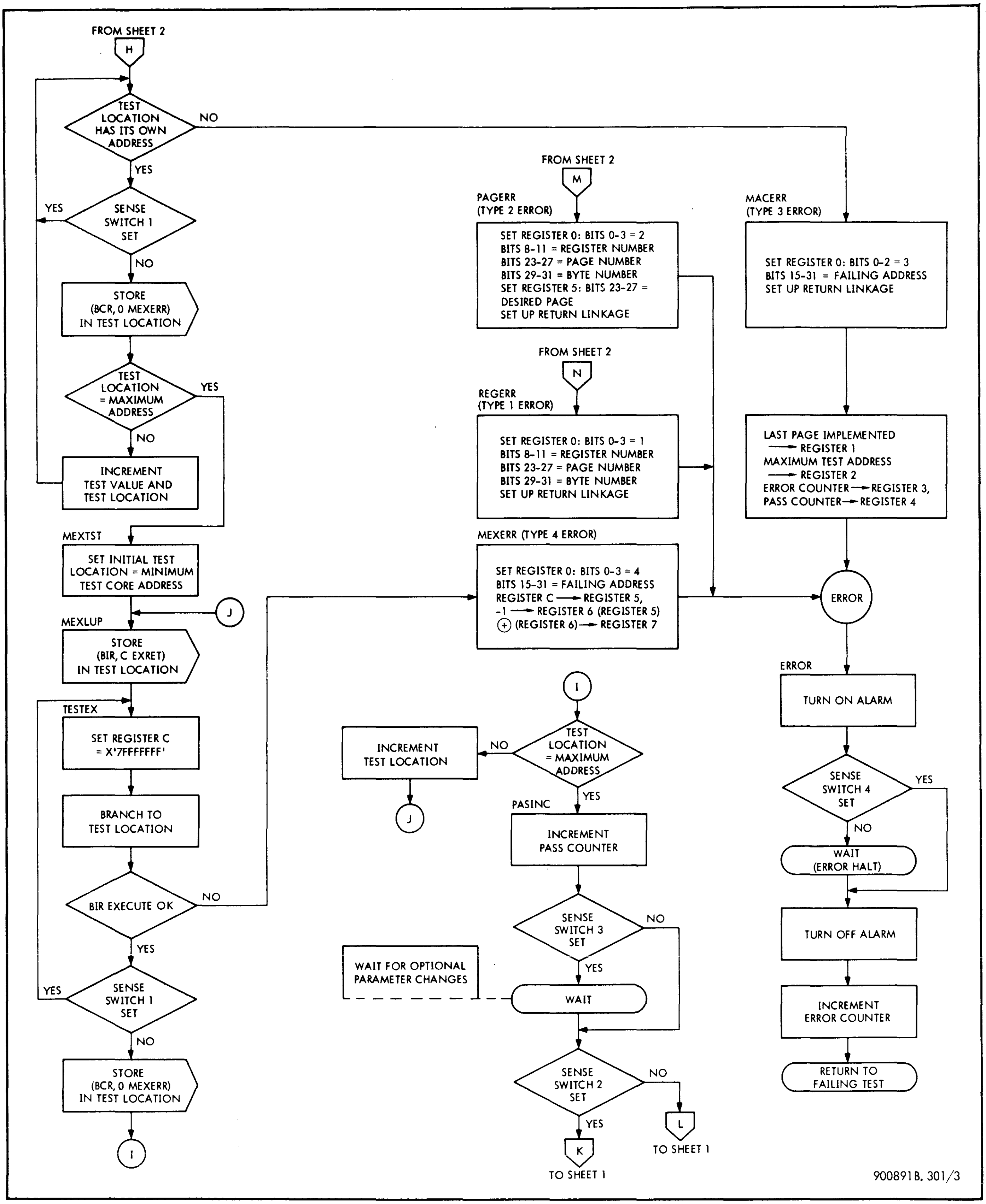
FROM SHEET 2
H
TEST LOCATION HAS ITS OWN ADDRESS
NO
YES
SENSE SWITCH 1 SET
YES
NO
STORE (BCR, 0 MEXERR) IN TEST LOCATION
TEST LOCATION = MAXIMUM ADDRESS
YES
NO
INCREMENT TEST VALUE AND TEST LOCATION
MEXTST
SET INITIAL TEST LOCATION = MINIMUM TEST CORE ADDRESS
J
MEXLUP
STORE (BIR, C EXRET) IN TEST LOCATION
TESTEX
SET REGISTER C = X'7FFFFFFF'
BRANCH TO TEST LOCATION
BIR EXECUTE OK
NO
YES
SENSE SWITCH 1 SET
YES
NO
STORE (BCR, 0 MEXERR) IN TEST LOCATION
I
FROM SHEET 2
M
PAGERR (TYPE 2 ERROR)
SET REGISTER 0: BITS 0-3 = 2
BITS 8-11 = REGISTER NUMBER
BITS 23-27 = PAGE NUMBER
BITS 29-31 = BYTE NUMBER
SET REGISTER 5: BITS 23-27 = DESIRED PAGE
SET UP RETURN LINKAGE
FROM SHEET 2
N
REGERR (TYPE 1 ERROR)
SET REGISTER 0: BITS 0-3 = 1
BITS 8-11 = REGISTER NUMBER
BITS 23-27 = PAGE NUMBER
BITS 29-31 = BYTE NUMBER
SET UP RETURN LINKAGE
MEXERR (TYPE 4 ERROR)
SET REGISTER 0: BITS 0-3 = 4
BITS 15-31 = FAILING ADDRESS
REGISTER C ⟶ REGISTER 5,
-1 ⟶ REGISTER 6 (REGISTER 5)
⊕ (REGISTER 6) ⟶ REGISTER 7
I
INCREMENT TEST LOCATION
NO
TEST LOCATION = MAXIMUM ADDRESS
J
YES
PASINC
INCREMENT PASS COUNTER
SENSE SWITCH 3 SET
NO
YES
WAIT FOR OPTIONAL PARAMETER CHANGES
WAIT
SENSE SWITCH 2 SET
NO
YES
K
TO SHEET 1
L
TO SHEET 1
MACERR (TYPE 3 ERROR)
SET REGISTER 0: BITS 0-2 = 3
BITS 15-31 = FAILING ADDRESS
SET UP RETURN LINKAGE
LAST PAGE IMPLEMENTED ⟶ REGISTER 1
MAXIMUM TEST ADDRESS ⟶ REGISTER 2
ERROR COUNTER ⟶ REGISTER 3,
PASS COUNTER ⟶ REGISTER 4
ERROR
ERROR
TURN ON ALARM
SENSE SWITCH 4 SET
YES
NO
WAIT (ERROR HALT)
TURN OFF ALARM
INCREMENT ERROR COUNTER
RETURN TO FAILING TEST
900891B. 301/3

Figure 3-1. Pattern Program Flow Chart (Sheet 3 of 3)

SECTION IV
PROGRAM LISTING

```
                                           SYSTEM   SIG7FDP
                                  * REVISION COO - REVISION DATE 1 OCT, 1968  PROGRAM REVISED TO RUN WITH
                                  * 1 REGISTER PAGE (PAGE 0), OR MORE. PREVIOUSLY IT REQUIRED MULTIPLE
                                  * PAGES,
                                  *
                                  *
                                  *
             0000000A             A        EQU      X'A'
             0000000B             B        EQU      X'B'
             0000000C             C        EQU      X'C'
             0000000D             D        EQU      X'D'
             0000000E             E        EQU      X'E'
             0000000F             F        EQU      X'F'
```

```
                                           PAGE
                                  *
                                  * PROGRAM CONTROL AND DISPLAY INFORMATION
                                  *
                                  *
                                  *        SENSE
                                  *        SWITCH   CONDITION   ACTION
                                  *        ------   ---------   ------
                                  *
                                  *        1        RESET       NORMAL OPERATION
                                  *                 SET         LOOP ON CURRENT TEST ( SS4 MUST BE ON
                                  *                             TO BYPASS THE ERROR-WAIT )
                                  *
                                  *        2        RESET       REPFAT TEST WITH CURRENT PAGE  LIMITS
                                  *                             AND/OR WITH CURRENT CORE PARAMETERS
                                  *
                                  *                 SET         RECALCULATE REGISTER-PAGE LIMITS
                                  *                             AND/OR CORE SIZE LIMITS
                                  *
                                  *        3        RESET       NORMAL OPERATION
                                  *                 SET         HALT TO CHANGE MEMORY LIMITS
                                  *                             AND/OR RP LIMITS ( SEE NEXT PAGE )
                                  *
                                  *        4        RESET       HALT ON ERRORS
                                  *                 SET         NO HALT ON FRRORS
```

```
                  PAGE
*
*
*       PCP INTERRUPT - REINITIALIZE AND EXECUTE PROGRAM.
*
* SENSE SWITCH 3 SET HALTS THE PROGRAM AT LOCATION '1AB'.  THE FOLLOWING
*          REGISTERS WILL DISPLAY THE SPECIFIED INFORMATION :-
*
*          REGISTER     CONTENTS
*          --------     --------
*
*          0            VALUE OF STARTING PAGE SHOWN IN BITS 20-27
*          1            MAXIMUM PAGES IMPLEMENTED''''''BITS 20-27
*          2            MINIMUM TEST CORE ADDRESS
*          3            MAXIMUM TEST CORE ADDRESS
*          4            PASS COUNT
*          5            ERROR COUNT
*
*          REGISTERS 0-3 MAY BE ALTERED PRIOR TO CLEARING THE HALT
```

```
                  PAGE
*
* AS A RESULT OF ANY  REGISTER-PAGE  ERRORS :-
* THE FOLLOWING ERROR INFORMATION IS DISPLAYED IN THE SPECIFIED
* REGISTERS SUBSEQUENT TO THE ERROR HALT AT LOCATION LABELLED 'ERHALT'
*
*       REGISTER   CONTENTS
*       --------   --------
*
*       0          ERROR IDENTIFIER AND LOCATION
*                    10R00PPB - ( TYPE 1 ) REGISTER PAGE ERROR
*                               FAULTY BYTE B IN REGISTER R OF PAGE PP
*
*                    20R00PPB - ( TYPE 2 ) MISADDRESSED PAGE PP - ERROR
*                               IN BYTE B OF REGISTER R
*
*       1          MAXIMUM PAGES IMPLEMENTED ( APPLIES TO ALL TYPES )
*       2          CURRENT PATTERN  (APPLIES TO ERROR-TYPE 1 ONLY )
*       3          RECEIVED PATTERN (APPLIES TO ERROR-TYPE 1 ONLY )
*       4          DIFFERING BITS IN R2 AND R4 (ERROR-TYPE 1 ONLY )
*
*       5          DESIRED PAGE --- (APPLIES TO ERROR-TYPE 2 ONLY )
*                   00000PPO - PAGE DESIRED WAS PP IN ADDRESSING
* 'PP' VALUE RANGES ARE 00-1F FOR SIGMA 7; 00-0F FOR SIGMA 5
*
* FOR MODULE LOCATIONS REFER TO TABLE '2-3' OF THE MANUAL
```

```
       84                                               PAGE
       85                              *
       86                              * AS A RESULT OF ANY MEMORY ERROR :-
       87                              * THE FOLLOWING ERROR INFORMATION IS DISPLAYED IN THE SPECIFIED
       88                              * REGISTERS SUBSEQUENT TO THE ERROR HALT AT LOCATION '1F3'
       89                              *
       90                              *        REGISTER   CONTENTS
       91                              *        --------   --------
       92                              *
       93                              *        0          ERROR IDENTIFIER AND LOCATION
       94                              *                     300YYYYY = ( TYPE 3 ) ACCESS ERROR AT CORE
       95                              *                                ADDRESS YYYYY
       96                              *                     400YYYYY = ( TYPE 4 ) EXACUTION ERROR AT CORE
       97                              *                                ADDRESS YYYYY
       98                              *
       99                              *        1          MAXIMUM PAGE IMPLEMENTED
      100                              *        2          MAXIMUM CORE ADDRESS
      101                              *        3          ACCUMULATIVE ERROR COUNT
      102                              *        4          NUMBER OF PROGRAM PASSES
      103                              *        5          ERRONEOUS RESULT
      104                              *        6          PREDETERMINED RESULT
      105                              *        7          DIFFERENCE BETWEEN R5 AND R6
      106                              *
      107                              *
      108                              *
      109                              *
      110                              * ADDITIONAL ERROR WAIT LOCATIONS:
      111                              *
      112                              * HALT AT LOCATION '06A' SIGNIFIES FAULTY REG.0, OR PAGES 1,2,OR3, OR
      113                              * REU NO.1 INTERFERENCE. FAULTY BITS INDICATED BY ZEROS IN REGISTER 0
      114                              *        FOR CORRECTIVE ACTION SEE PARAGRAPH  2-4  OF THE MANUAL
      115                              *
      116                              *        HALT AT LOCATION '071' SIGNIFIES FAULTY R0 AT BITS SHOWN BY 1
      117                              *
      118                              *        HALTS AT ALL ODD LOCATIONS '24F' TO '277' INDICATE SPECIFIC
      119                              *        SPURIOUS TRAP OR INTERRUPT.  SEE LISTING FOR IDENTIFICATION
```

```
      120                                                PAGE
      121   01 00040                                     ORG      X'40'
            01 00040
      122                              *
      123                              * TRAP LOCATIONS
      124                              *
      125   01 00040    0F4001EC                         XPSD,4   NONOP
      126   01 00041    0F0001FA                         XPSD,0   UNIMP
      127   01 00042    0F0001FE                         XPSD,0   STACK
      128   01 00043    0F000202                         XPSD,0   OFLO
      129   01 00044    0F000206                         XPSD,0   FLOAT
      130   01 00045    0F00020A                         XPSD,0   DEC
      131   01 00046    0F00020E                         XPSD,0   TIMER
      132   01 00047    0F000222                         XPSD,0   TRAPUN
      133   01 00048    0F000212                         XPSD,0   CALL1
      134   01 00049    0F000216                         XPSD,0   CALL2
      135   01 0004A    0F00021A                         XPSD,0   CALL3
      136   01 0004B    0F00021E                         XPSD,0   CALL4
      137   01 0004C    0F000222                         XPSD,0   TRAPUN
      138   01 0004D    0F000222                         XPSD,0   TRAPUN
      139   01 0004E    0F000222                         XPSD,0   TRAPUN
      140   01 0004F    0F000222                         XPSD,0   TRAPUN
```

```
                                   *
                                   * INTERRUPT LØCATIØNS
                                   *
 01 00050     0F000226                     XPSD,0    PØWØN
 01 00051     0F00022A                     XPSD,0    PØWØFF
 01 00052     3300022E                     MTW,0     PULSE1
 01 00053     3300022F                     MTW,0     PULSE2
 01 00054     33000230                     MTW,0     PULSE3
 01 00055     33000231                     MTW,0     PULSE4
 01 00056     0F000232                     XPSD,0    MEMPAR
 01 00057     0F000236                     XPSD,0    INTUN
 01 00058     0F00023A                     XPSD,0    CØUNT1
 01 00059     0F00023E                     XPSD,0    CØUNT2
 01 0005A     0F000242                     XPSD,0    CØUNT3
 01 0005B     0F000246                     XPSD,0    CØUNT4
 01 0005C     0F00024A                     XPSD,0    INØUT
 01 0005D     0F00024E                     XPSD,0    PCPINT
 01 0005E     0F000236                     XPSD,0    INTUN
 01 0005F     0F000236                     XPSD,0    INTUN
```

```
                                   *
                                   * REGISTER 0 TEST - ENTRY FRØM PRØGRAM LØAD ØR PCP INTERRUPT
                                   *
 01 00060     2F0002C1             START   LRP       ZFRØ             PAGE 0
 01 00061     3200029D             REUTST  LW,0      MINS1            R0=-1
 01 00062     4800029D                     EØR,0     MINS1            IS R0=-1, TEST FØR REU AND R0
 01 00063     69300066                     BCS,3     $+3              NØ, GØ WAIT
 01 00064     32000000 A                   LW,0      0                IS R0=0
 01 00065     6830006B                     BCR,3     REUØK            YES, NØ REU INTERFERENCE
 01 00066     3200029D                     LW,0      MINS1            R0=-1
 01 00067     6C000010 A                   RD,0      X'10'            SS1 SET
 01 00068     69800061                     BCS,8     REUTST           YES, TEST REU AND R0 AGAIN
 01 00069     2E000000 A                   WAIT                       NØ, EITHER REU ØR RØ FAULTY
                                   *
                                   * FAULTY R0 ØR REU -  SEE PAGE 5 ØF THIS LISTING UNDER ADDITIØNAL ERRØR
                                   *
 01 0006A     68000061                     B         REUTST           LØØP UNTIL BØTH REU AND R0 ARE ØK
 01 0006B     22000000 A           REUØK   LI,0      0                TEST R0 FØR ØPEN CIRCUIT
 01 0006C     48000000 A                   EØR,0     0                IS R0=0
 01 0006D     68300072                     BCR,3     R0GØØD           YES, PRØCEED WITH THE TEST
 01 0006E     6C000010 A                   RD,0      X'10'            SS1 SET
 01 0006F     6980006B                     BCS,8     REUØK            YES, TEST R0 FØR ØPEN CIRCUITS AGAIN
 01 00070     2E000000 A                   WAIT                       R0 FAULTY AT BITS INDICATED BY 1
                                   *
                                   * FAULTY R0 -  SEE PAGE 5 ØF THIS LISTING UNDER 'ADDITIØNAL ERRØR WAITS'
                                   *
 01 00071     6800006B                     B         REUØK
```

```
                                              PAGE
                                       * INITIALIZATION PROCEDURE
                                       *
       01 00072     321002A0           ROGOOD   LW,1     ARMINT             ARM AND ENABLE PCP, PARITY INTERUPT
       01 00073     6D101200 A                  WD,1     X'1200'
       01 00074     320002A5                    LW,0     HALT
       01 00075     35000011 A                  STW,0    X'11'
       01 00076     32000289                    LW,0     BRST
       01 00077     35000026 A                  STW,0    X'26'
       01 00078     320002C1                    LW,0     ZERO               CLEAR PASS COUNTER
       01 00079     350002BA                    STW,0    PASCTR
       01 0007A     350002BB                    STW,0    PASTAG             CLEAR INITIAL PASSING TAG
       01 0007B     320002B4                    LW,0     ONE                CLEAR ERROR COUNTER TO 1
       01 0007C     350002A4                    STW,0    ERRCTR
                                       *
                                       * PROCEDURE TO CALCULATE THE REGISTER PAGE LIMITS
                                       *
       01 0007D     320002BC           RESET    LW,0     REGLIM             BEGIN TO CALCULATE NO. OF PAGES
       01 0007E     350002B6                    STW,0    PAGE
       01 0007F     350002B7                    STW,0    PAGEP                                                   *C
       01 00080     65000081           PSCAN    BIR,0    $+1                INCREMENT PAGE COUNT
       01 00081     480002BC                    EOR,0    REGLIM
       01 00082     350002B6                    STW,0    PAGE               PAGE=00000PPF
       01 00083     480002A9                    EOR,0    MAXPAG             ARE ALL 32 PAGES IMPLEMENTED
       01 00084     6830008E                    BCR,3    ENDPAG             YES
       01 00085     320002C1                    LW,0     ZERO
       01 00086     2F0002B6                    LRP      PAGE
       01 00087     32000292                    LW,0     FIVES              FIVES =X'555555'
       01 00088     32100000 A                  LW,1     0                  IS THIS PAGE IMPLEMENTED
       01 00089     6820008E                    BCR,2    ENDPAG             NO
       01 0008A     2F0002C1                    LRP      ZERO               YES, INVESTIGATE NEXT PAGE,BY PAGE 0
       01 0008B     320002B6                    LW,0     PAGE
       01 0008C     350002B7                    STW,0    PAGEP
       01 0008D     68000080                    B        PSCAN              SCAN NEXT PAGE
       01 0008E     2F0002C1           ENDPAG   LRP      ZERO               USE PAGE 0
       01 0008F     320002B7                    LW,0     PAGEP
       01 00090     480002BC                    EOR,0    REGLIM             MASK OUT 4 LEAST SIG. DIGITS
```

```
       01 00091     350002B9                    STW,0    PAGLIM             STORE PAGE LIMIT
```

```
                                  *
                                  * SELECT THE PAGE TØ BE TESTED
                                  *
 01 00092    320002C2      TESTPG  LW,0    7FSZRØ          RØ=X'FFFFFFFØ'
 01 00093    350002B6              STW,0   PAGE            INITILIZE PAGE CØUNTER
 01 00094    320002B6      NEWPAG  LW,0    PAGE            GET CURRENT PAGE
 01 00095    480002B9              EØR,0   PAGLIM          IS THIS LAST PAGE
 01 00096    683000F5              BCR,3   TESTPP          YES, GØ TØ DØ PAGE ADDRESSING TEST
 01 00097    320002B6              LW,0    PAGE            GET CURRENT PAGE AGAIN
 01 00098    480002BC              EØR,0   REGLIM          EØR WITH F
 01 00099    6500009A              BIR,0   $+1             INCREMENT PAGE CØUNTER
 01 0009A    350002B6              STW,0   PAGE            STØRE NEW PAGE CØUNTER
                                  *
                                  * SELECT TEST PATTERNS IN CØNSECUTIVE ØRDER
                                  *
 01 0009B    2F0002C1      GETPAT  LRP     ZERØ            RETURNTØ PAGE ZERØ
 01 0009C    3200028C              LW,0    PATSCT          PATSCT=LW,0 PATTRN
 01 0009D    350000A5              STW,0   PATNØW          SET PATTERN SELECTØR CØDE
 01 0009E    3200029D              LW,0    MINS1           ALL 1'S
 01 0009F    3500028E              STW,0   PATCNT          SET PATTERN CØUNTER =-1
 01 000A0    3200028E      NXTPAT  LW,0    PATCNT          GET PATTERN CØUNTER
 01 000A1    650000A2              BIR,0   $+1             INCREMENT PATTERN CØUNTER
 01 000A2    3500028E              STW,0   PATCNT
 01 000A3    4800028B              EØR,0   PATLIM          LAST PATTERN
 01 000A4    68300094              BCR,3   NEWPAG
                                  *
                                  * EXECUTE ' LW,0 PATTRN + PATCNT ' TØ PØINT TØ PRØPER PATTERN
                                  *
 01 000A5    00000000 A    PATNØW  DATA    0               SELECT TEST PATTERN
 01 000A6    3500028F              STW,0   PATSTØ          PUT CURRENT TEST PATTERN IN MEMØRY
 01 000A7    4800029D              EØR,0   MINS1           FIND 1'S CØMPLIEMENT ØF TEST PATTERN
 01 000A8    35000290              STW,0   CMPATR          STØRE IT IN MEMØRY
 01 000A9    320000A5              LW,0    PATNØW
 01 000AA    650000AB              BIR,0   $+1
 01 000AB    350000A5              STW,0   PATNØW
```

```
                                  *
                                  * LØAD CURRENT REGISTER WITH PATTERN
                                  *
 01 000AC    320002C1      DUMPAT  LW,0    ZERØ
 01 000AD    35000297              STW,0   R               R=0
 01 000AE    35000298              STW,0   RR              R-CØUNTER = 0
 01 000AF    32000296              LW,0    LØDPAT          LØDPAT = LW,0 PATS TØ
 01 000B0    350000B4              STW,0   SELECT          SET REGISTER SELECTØR CØDE
 01 000B1    3200029E              LW,0    STRPAT          STRPAT = STW,0 TEMP
 01 000B2    350000CD              STW,0   STØRE           SET REGISTER STØRAGE CØDE
 01 000B3    2F0002B6      NEXTRG  LRP     PAGE            SET PRINTER TØ CURRENT PAGE
 01 000B4    00000000 A    SELECT  DATA    0               LØAD PATTERN TØ CURRENT REGISTER R
                                  *
                                  * LØAD ALL REGISTERS ( EXCEPT CURRENT REG.) ØF THE CURRENT PAGE WITH
                                  * CØMPLEMENT ØF THE CURRENT PATTERN
                                  *
 01 000B5    2F0002C1              LRP     ZERØ            RETURN TØ PAGE 0
 01 000B6    32000298      REGLØD  LW,0    RR              GET R-CØUNTER
 01 000B7    48000297              EØR,0   R               IS R-CØUNTER = TØ CURRENT REG. R
 01 000B8    683000BF              BCR,3   PRESRG          YES, SKIP LØADING IT WITH CMPATR
 01 000B9    32000294              LW,0    LØADRG          LØADRG = LW,0 CMPATR
 01 000BA    48000298              EØR,0   RR              MØDFY R FIELD
 01 000BB    350000BD              STW,0   REGMØD          SET REGISTER SELECTØR CØDE
 01 000BC    2F0002B6              LRP     PAGE            ADVANCE TØ CURRENT PAGE
 01 000BD    00000000 A    REGMØD  DATA    0               LØAD CØMPL. ØF PATTERN TØ REG. R
 01 000BE    2F0002C1              LRP     ZERØ            RETURN TØ PAGE 0
 01 000BF    32000298      PRESRG  LW,0    RR              GET R-CØUNTER
 01 000C0    48000291              EØR,0   FIVEFS          MØDIFY LØWER BYTES FØR INCREMENTING
 01 000C1    650000C2              BIR,0   $+1             INCREMENT R-CØUNTER
 01 000C2    35000298              STW,0   RR              STØRE IT BACK
 01 000C3    4800029A              EØR,0   RMAX            WAS THIS LAST REGISTER
 01 000C4    683000C6              BCR,3   TSTREG          YES, GØ TEST THE CURRENT TEST-REG.
 01 000C5    680000B6              B       REGLØD          REPEAT TØ LØAD NEXT REGISTER
```

```
     297                                           PAGE
     298                                     *
     299                                     * EXAMINE THE TEST REGISTER FOR CONTAINING THE CURRENT PATTERN
     300                                     *
     301    01 000C6     22000000 A   TSTREG   LI,0      0
     302    01 000C7     35000298              STW,0     RR                    INITIALIZE R-COUNTER
     303    01 000C8     32000286              LW,0      PAGE                  IS THIS PAGE 0
     304    01 000C9     693000CC              BCS,3     EXAMIN                YES, MAKE TEST
     305    01 000CA     32000297              LW,0      R                     IS THIS R0 OF PAGE 0
     306    01 000CB     683000D2              BCR,3     REGOOD                YES, DO NOT TEST R0
     307    01 000CC     2F000286     EXAMIN   LRP       PAGE                  RETURN TO TEST PAGE
     308    01 000CD     00000000 A   STORE    DATA      0                     STORE REG. R IN LOCATION TEMP
     309    01 000CE     2F0002C1              LRP       ZERO                  RETURN TO PAGE 0
     310    01 000CF     3200029F              LW,0      TEMP                  FETCH STORED DATA
     311    01 000D0     4800028F              EOR,0     PATSTO                COMPARE WITH CURRENT PATTERN
     312    01 000D1     693000DF              BCS,3     BYTERR                UNEQUAL   GO TO FIND FAULTY BYTE
     313    01 000D2     32000297     REGOOD   LW,0      R                     GET R
     314    01 000D3     48000291              EOR,0     FIVEFS                MODIFY TO INCREMENT BY OVERFLOW
     315    01 000D4     650000D5              BIR,0     $+1                   INCREMENT R
     316    01 000D5     35000297              STW,0     R                     STORE NEW VALUE OF R
     317    01 000D6     4800029A              EOR,0     RMAX                  LAST REGISTER
     318    01 000D7     683000A0              BCR,3     NXTPAT                YES, REPEAT CURRENT PAGE TEST WITH
     319                              *                                        NEXT PATTERN
     320    01 000D8     32000297              LW,0      R
     321    01 000D9     48000296              EOR,0     LODPAT                LODPAT = LW,0 PATSTO
     322    01 000DA     350000B4              STW,0     SELECT                MODIFY LOAD LOCATION FOR NEXT REG.
     323    01 000DB     32000297              LW,0      R
     324    01 000DC     4800029E              EOR,0     STRPAT                STRPAT= STW,0 TEMP
     325    01 000DD     350000CD              STW,0     STORE                 MODIFY STORE LOCATION FOR NEXT REG.
     326    01 000DE     680000B3              B         NEXTRG                GO TEST NEXT REGISTER
```

```
     327                                           PAGE
     328                                     *
     329                                     * FIND FAULTY BYTE IN THE SPECIFIED REGISTER
     330                                     *
     331    01 000DF     320002C1     BYTERR   LW,0      ZERO
     332    01 000E0     35000288              STW,0     BYTCNT                INITIALIZE THE BYTE COUNTER
     333    01 000E1     32000286              LW,0      BYTMOD                BYTMOD = LW,0 BYTO
     334    01 000E2     350000F3              STW,0     BYTNOW                SET BYE MODIFYING CODE
     335    01 000E3     00000000 A   BYTNOW   DATA      0                     GET PROPER BYTES MASK
     336    01 000E4     3500029C              STW,0     MASK                  STORE IN MASK
     337    01 000E5     3200029F              LW,0      TEMP                  GET STORED BYTE
     338    01 000E6     4800028F              EOR,0     PATSTO                TEST THE BYTE WITH ORIGINAL BYTE
     339    01 000E7     4B00029C              AND,0     MASK                  DISCARD OTHER INTERFERING BYTES
     340    01 000E8     693001B8              BCS,3     REGERR                ERROR,  GO TO ERROR ROUTINE
     341    01 000E9     32000288     RETBYT   LW,0      BYTCNT                NO ERROR
     342    01 000EA     650000EB              BIR,0     $+1                   INCREMENT BYTE COUNTER
     343    01 000EB     35000288              STW,0     BYTCNT
     344    01 000EC     48000284              EOR,0     BYTMAX                LAST BYTE
     345    01 000ED     683000F2              BCR,3     TSTSS1                YES, TEST SS1 FOR REPETITION
     346    01 000EE     320000F3              LW,0      BYTNOW                NO
     347    01 000EF     650000F0              BIR,0     $+1                   MODIFY LOCATION TO GET NEXT BYTE
     348    01 000F0     350000E3              STW,0     BYTNOW
     349    01 000F1     680000E3              B         BYTNOW                GO TEST NEXT BYTE
     350                                     *
     351                                     * TEST SWITCH 1,  SS = SET IF TESTING THE SAME FAULTY REGISTER DESIRED
     352                                     *
     353    01 000F2     6C000010 A   TSTSS1   RD,0      X'10'                 IS SS1 SET
     354    01 000F3     698000DF              BCS,8     BYTERR                YES, REPEAT TESTING SAME REGISTER
     355    01 000F4     680000D2              B         REGOOD                NO, ADVANCE TO TEST NEXT REGISTER
```

```
356
357                                        *
358                                        * TEST FØR CØRRECT ADDRESSING ØF THEHE IMPLEMENTED PAGES
359                                        *
360   01 000F5      2F0002C1      TESTPP   LRP       ZERØ             SET RP TØ PAGE ZERØ
361   01 000F6      32000289               LW,0      PAGLIM           IS ØNLY PAGE 0  IMPLEMENTED          *C
362   01 000F7      68300156               BCR,3     CØRSIZ           YES,   GØ TØ CØRE TEST               *C
363   01 000F8      320002C1               LW,0      ZERØ
364   01 000F9      35000286               STW,0     PAGE             INITIALIZE PAGE CØUNTER
365   01 000FA      320002C1      FILL1S   LW,0      ZERØ
366   01 000FB      35000297               STW,0     R                R=0
367   01 000FC      32000295               LW,0      LDMNS1           SET CØDE TØ LØAD REGISTERS *-1
368   01 000FD      350000FF               STW,0     MØDREG
369   01 000FE      2F000286      LØDREG   LRP       PAGE             SET CURRENT PAGE
370                                        *
371                                        * EXACUTE ' LW,R  *-1 ' IN ØRDER TØ LØAD ALL REGS. ØF CURRENT PAGE *-1
372                                        *
373   01 000FF      00000000 A    MØDREG   DATA      0
374   01 00100      2F0002C1               LRP       ZERØ             BACK TØ PAGE 0
375   01 00101      32000297               LW,0      R                FETCH R
376   01 00102      48000291               EØR,0     FIVEFS           MØDIFY LØWER BITS FØR INCREMENTATIØN
377   01 00103      65000104               BIR,0     $+1              INCREMENT R BY ØVERFLØW
378   01 00104      35000297               STW,0     R                STØRE NEW R
379   01 00105      4800029A               EØR,0     RMAX             LAST REGISTER IN CURRENT PAGE
380   01 00106      6830010B               BCR,3     FLPAGE           YES, FLIP THE PAGE
381   01 00107      32000297               LW,0      R
382   01 00108      48000295               EØR,0     LDMNS1           LDMNS1= LW,0 MINS1
383   01 00109      350000FF               STW,0     MØDREG           MØDIFY LØCATIØN TØ SET NEXT REG. *-1
384   01 0010A      680000FE               B         LØDREG           LØAD NEXT R *-1
```

```
385
386                                        *
387                                        * FLIP TØ LØAD REGISTERS ØF THE PAGE *-1
388                                        *
389   01 0010B      2F0002C1      FLPAGE   LRP       ZERØ             RETURN TØ PAGE 0
390   01 0010C      32000286               LW,0      PAGE             GET PAGE CØUNTER
391   01 0010D      48000289               EØR,0     PAGLIM           LAST PAGE
392   01 0010E      68300114               BCR,3     FULPGS           YES
393   01 0010F      32000286               LW,0      PAGE             GET PAGE CØUNTER
394   01 00110      4800028C               EØR,0     REGLIM           MØDIFY TØ INCREMENT BY ØVERFLØW
395   01 00111      65000112               BIR,0     $+1              INCREMENT PAGE CØUNTER
396   01 00112      35000286               STW,0     PAGE             STØRE NEW VALUE ØF THE PAGE CØUNTER
397   01 00113      680000FA               B         FILL1S           NØ, LØAD NEXT PAGE ALL REGS. *-1
398   01 00114      320002C1      FULPGS   LW,0      ZERØ
399   01 00115      35000286               STW,0     PAGE             INITIALIZE PAGE CØUNTER
400   01 00116      35000293               STW,0     FLPCNT           INITIALIZE PAGE FLIP
401   01 00117      2F000286      BEGPAG   LRP       PAGE             SET RP TØ CURRENT PAGE
402   01 00118      320002C1               LW,0      ZERØ             CLEAR REGISTER 0
403   01 00119      328002C1               LW,8      ZERØ             CLEAR REGISTER 8
```

```
      404                                                PAGE
      405                                        *
      406                                        * FLIP PAGES AND CHECK CØNTENTS FØR ANY CHANGE DUE TØ MISSADRESSING
      407                                        *
      408    01 0011A     2F0002C1     FLIPAG    LRP       ZERØ              RETURN TØ PAGE 0
      409    01 0011B     32000293               LW,0      FLPCNT
      410    01 0011C     480002B9               EØR,0     PAGLIM            LAST PAGE
      411    01 0011D     6830012D               BCR,3     PAGNXT            YES, PRØCEED TØ NEXT PAGE
      412    01 0011E     32000293               LW,0      FLPCNT            GET PAGE FLIP CØUNTER
      413    01 0011F     480002BC               EØR,0     REGLIM            MØDIFY TØ INCREMENT
      414    01 00120     65000121               BIR,0     $+1               INCREMENT BY ØVERFLØW
      415    01 00121     35000293               STW,0     FLPCNT            STØRE NEW VALUE ØF PAGE FLIP
      416    01 00122     320002C1               LW,0      ZERØ
      417    01 00123     2F000293               LRP       FLPCNT            RETURN TØ SAME PAGE FLIPED
      418    01 00124     4800029D               EØR,0     MINS1             CLEAR REGISTER 0
      419    01 00125     69300137               BCS,3     ERROT7            MISADDRESSING IN REGS 0-7 THIS PAGE
      420    01 00126     3200029D               LW,0      MINS1             RESTØRE REGISTER 0 =-1
      421    01 00127     4880029D               EØR,8     MINS1             CLEAR REGISTER 8
      422    01 00128     6930013D               BCS,3     ERR8TF            MISADDRESSING IN REGS 8-F THIS PAGE
      423    01 00129     2F000293     FRESHP    LRP       FLPCNT
      424    01 0012A     3280029D               LW,8      MINS1
      425    01 0012B     3200029D               LW,0      MINS1
      426    01 0012C     6800011A               B         FLIPAG            GØ TØ FLIP THE PAGE AND TEST
      427    01 0012D     320002B6     PAGNXT    LW,0      PAGE              SET RP TØ CURRET PAGE
      428    01 0012E     480002BC               EØR,0     REGLIM            MØDIFY PAGE TØ INCREMENT
      429    01 0012F     65000130               BIR,0     $+1               INCREMENT THE PAGE
      430    01 00130     350002B6               STW,0     PAGE              STØRE NEW VALUE ØF PAGE
      431    01 00131     480002B9               EØR,0     PAGLIM            LAST PAGE
      432    01 00132     68300156               BCR,3     CØRSIZ            YES, GØ TØ CØRE TEST
      433    01 00133     320002B6               LW,0      PAGE              NØ, FETCH PAGE TESTED
      434    01 00134     35000293               STW,0     FLPCNT            STØRE IT TØ START FLIPPING FRØM HERE
      435    01 00135     3200029D               LW,0      MINS1             RESØTRE REGISTER 0 TØ -1
      436    01 00136     68000117               B         BEGPAG            PRØCEED TØ TEST NEXT PAGE
```

```
      437                                                PAGE
      438                                        *
      439                                        * DISTINGUISH BETWEEN REGISTERS 0-7 AND 8-F IN ADDRESSING ERRØRS
      440                                        *
      441    01 00137     350002BD     ERROT7    STW,0     SAVERR            SAVE FAULTY REGISTER
      442    01 00138     3200029D               LW,0      MINS1             RESTØRE R0 =-1
      443    01 00139     2F0002C1               LRP       ZERØ              BACK TØ PAGE ZERØ
      444    01 0013A     320002C1               LW,0      ZERØ              R0=0
      445    01 0013B     35000297               STW,0     R                 SELECT R0
      446    01 0013C     68000141               B         PFAULT
      447    01 0013D     350002BD     ERR8TF    STW,0     SAVERR            SAVE FAULTY REGISTER
      448    01 0013E     2F0002C1               LRP       ZERØ              BACK TØ PAGE 0
      449    01 0013F     32000299               LW,0      R8                SELECT R8
      450    01 00140     35000297               STW,0     R
      451    01 00141     320002C1     PFAULT    LW,0      ZERØ
      452    01 00142     35000288               STW,0     BYTCNT            BYTE CØUNT  0
      453    01 00143     320002B6               LW,0      BYTMØD            BYTMØD=LW,0 BYTE0
      454    01 00144     35000145               STW,0     BYTCUR
      455    01 00145     00000000 A   BYTCUR    DATA      0                 SELECT UNDESIRED BYTES
      456    01 00146     3500029C               STW,0     MASK              STØRE IN LØCATIØN MASK
      457    01 00147     320002BD               LW,0      SAVERR            FETCH FAULTY REGISTER
      458    01 00148     4800029C               AND,0     MASK              SELECT DESIRED BYTE
      459    01 00149     693001C5               BCS,3     PAGERR            ANY DIGIT =1   IF YES NØTE AS ERRØR
      460    01 0014A     32000288     RETPAG    LW,0      BYTCNT            NØ
      461    01 0014B     6500014C               BIR,0     $+1               INCREMENT BYTE CØUNTER
      462    01 0014C     35000288               STW,0     BYTCNT
      463    01 0014D     48000284               EØR,0     BYTMAX            LAST BYTE
      464    01 0014E     68300153               BCR,3     SS1TST            CHECK SS1
      465    01 0014F     32000145               LW,0      BYTCUR
      466    01 00150     65000151               BIR,0     $+1               INCREMENT LØCATIØN TØ TEST NEXT BYTE
      467    01 00151     35000145               STW,0     BYTCUR
      468    01 00152     68000145               B         BYTCUR            GØ TØ TEST NEXT BYTE
      469    01 00153     6C000010 A   SS1TST    RD,0      X'10'             SS1 SET
      470    01 00154     69800141               BCS,8     PFAULT            YES, REPEAT TEST FØR THE SAME PAGE
      471    01 00155     68000129               B         FRESHP            NØ,
                                PAGE
                                *
                                * CALCULATE MAXIMUM CØRE ADDRESS
                                *
01 00156   2F0002C1     CØRSIZ  LRP      ZERØ          RESET PAGE PØINTER
01 00157   320002BB             LW,0     PASTAG        IS THIS THE FIRST PASS
01 00158   6830015B             BCR,3    $+3           YES, CALCULATE CØRE SIZE
01 00159   6C000010 A           RD,0     X'10'         IS SS2 SET
01 0015A   68400168             BCR,4    CØRFIL        NØ, USE CURRENT CØRE SIZE
01 0015B   321002B3             LW,1     N300          YES, RECALCULATE CØRE SIZE
01 0015C   320002AC             LW,0     MEMFIL
01 0015D   6500015E     SETFIL  BIR,0    $+1           INCREMENT STW ADDRESS
01 0015E   3500015F             STW,0    $+1
01 0015F   00000000 A           DATA     0             EXECUTE STW,1  (R0)+1
01 00160   6510015D             BIR,1    SETFIL        INCREMENT FILL DATA
01 00161   321002A8             LW,1     MAXCØR        128K CØRE IMPLEMENTED
01 00162   481002A8     MAXMEM  AND,1    MAXCØR        TRAP RETURN - MASK ØUT NØN-ADDRESS
01 00163   351002AB             STW,1    MEMAX         SAVE MAX ADDRESS
01 00164   321002B5             LW,1     P300
01 00165   351002AD             STW,1    MEMIN         SAVE MIN TEST ADDRESS
01 00166   32000292             LW,0     FIVES
01 00167   350002BB             STW,0    PASTAG        STØRE NØN-ZERØ NØ. IN PASTAG
                                *
                                * SET EACH CØRE LØCATIØN EQUAL TØ ITS ØWN ADDRESS
                                *
01 00168   328002AF     CØRFIL  LW,8     MEMSTØ
01 00169   488002AD             EØR,8    MEMIN         ADD MIN TEST ADDRESS
01 0016A   326002AD             LW,6     MEMIN         INITIAL ADDRESS AND TEST VALUE
01 0016B   3580016C     CØRLØD  STW,8    $+1
01 0016C   00000000 A           DATA     0             EXECUTE STW,6 TEST ADDRESS
01 0016D   329002AB             LW,9     MEMAX
01 0016E   48900006 A           EØR,9    6             TEST VALUE = MAX ADDRESS
01 0016F   68300173             BCR,3    MACTST        YES
01 00170   65600171             BIR,6    $+1           NØ, INCREMENT TEST VALUE
01 00171   65800172             BIR,8    $+1           INCREMENT TEST ADDRESS
01 00172   6800016B             BCR,0    CØRLØD
```

```
                                PAGE
                                *
                                * TEST CØRE MEMØRY BETWEEN DEFINED LIMITS FØR CØNTENT EQUAL TØ ADDRESS
                                *
01 00173   32A002BE     MACTST  LW,A     STØBCR
01 00174   48A002AD             EØR,A    MEMIN         ADD MIN TEST ADDRESS
01 00175   32B002A3             LW,B     BTØERR        ERRØR BRANCH FØR MEM EXECUTE TEST
01 00176   328002AE             LW,8     MEMLØD
01 00177   488002AD             EØR,8    MEMIN         ADD TEST ADDRESS
01 00178   326002AD             LW,6     MEMIN         INITIAL TEST VALUE
01 00179   3580017A     MACLUP  STW,8    $+1
01 0017A   00000000 A           DATA     0             EXECUTE  LW,5 TEST ADDRESS
01 0017B   32700006 A           LW,7     6             PREDETERMINED VALUE
01 0017C   48700005 A           EØR,7    5             PREDETERMINED = ACTUAL
01 0017D   6830017F             BCR,3    $+2           YES
01 0017E   680001CF             BCR,0    MACERR        NØ, MEMØRY ACCESS ERRØR
01 0017F   6C000010 A   MAERET  RD,0     X'10'         SS1 SET
01 00180   6980017A             BCS,8    MACLUP+1      YES, REPEAT TEST FØR THIS ADDRESS
01 00181   35A00182             STW,A    $+1           NØ,
01 00182   00000000 A           DATA     0             STØRE (BCR,0 MEXERR)FØR EXECUTE TEST
01 00183   329002AB             LW,9     MEMAX
01 00184   48900006 A           EØR,9    6             TEST VALUE = MAX ADDRESS
01 00185   6830018A             BCR,3    MEXTST        YES, PRØCEED TØ NEXT TEST
01 00186   65600187             BIR,6    $+1           INCREMENT TEST VALUE
01 00187   65800188             BIR,8    $+1           INCREMENT TEST ADDRESS
01 00188   65A00189             BIR,A    $+1           INCREMENT REFILL ADDRESS
01 00189   68000179             BCR,0    MACLUP
```

```
                                           PAGE
                                   *
                                   * TEST INSTRUCTION EXECUTION FROM EACH LOCATION WITHIN MEMORY LIMITS
                                   *
01 0018A    329002A3     MEXTST    LW,9      BTOERR           LOAD BRANCH TO ERROR
01 0018B    328002C0               LW,8      STORE9           LOAD STORE R9
01 0018C    488002AD               EOR,8     MEMIN            ADD MIN TEST ADDRESS
01 0018D    32A002A1               LW,A      BIRLOD           LOAD TEST BIR
01 0018E    32B002BF               LW,B      STOBIR           LOAD STORE RA
01 0018F    48B002AD               EOR,B     MEMIN            ADD MIN TEST ADDRESS
01 00190    32D002A2               LW,D      BTOBIR           LOAD BRANCH TO TEST ADDRESS
01 00191    48D002AD               EOR,D     MEMIN            ADD MIN TEST ADDRESS
01 00192    35B00193     MEXLUP    STW,B     $+1
01 00193    00000000 A             DATA      0                EXECUTE STW,A TEST ADDRESS
01 00194    32C002AA     TESTEX    LW,C      MAXPOS           RC = LARGEST POSITIVE VALUE
01 00195    35D00196               STW,D     $+1
01 00196    00000000 A             DATA      0                EXECUTE BCR,0 TEST ADDRESS
01 00197    6C000010 A   EXRET     RD,0      X'10'            SUCCESSFUL EXECUTION RETURN
01 00198    69800194               BCS,8     TESTEX           SS1 SET - REPEAT TEST FOR THIS ADDR
01 00199    3580019A               STW,8     $+1
01 0019A    00000000 A             DATA      0                REPLACE BIR WITH BCR TO ERROR
01 0019B    32E00008 A             LW,E      8                CURRENT TEST LOCATION
01 0019C    4BE002A8               AND,E     MAXCOR           ADDRESS MASK
01 0019D    48E002AB               EOR,E     MEMAX            CURRENT ADDRESS = MEMORY LIMIT
01 0019E    683001A3               BCR,3     PASINC           YES, EXECUTION TEST COMPLETED
01 0019F    658001A0               BIR,8     $+1              NO, TEST NEXT LOCATION
01 001A0    65B001A1               BIR,B     $+1
01 001A1    65D001A2               BIR,D     $+1
01 001A2    68000192               BCR,0     MEXLUP
```

```
                                           PAGE
                                   *
                                   * END OF PROGRAM PASS. INCREMENT PASS COUNTER AND DISPLAY MEMORY
                                   * PARAMETERS FOR ALTLRATION IF REQUESTED.
                                   *
01 001A3    324002BA     PASINC    LW,4      PASCTR
01 001A4    654001A5               BIR,4     $+1
01 001A5    354002BA               STW,4     PASCTR
01 001A6    320002B8               LW,0      PAGBEG
01 001A7    6C000010 A             RD,0      X'10'            SS3 SET
01 001A8    682001B1               BCR,2     REITER           NO
01 001A9    321002B9               LW,1      PAGLIM
01 001AA    322002AD               LW,2      MEMIN
01 001AB    323002AB               LW,3      MEMAX
01 001AC    325002A4               LW,5      ERRCTR
01 001AD    2E000000 A             WAIT                       YES, HALT FOR CHANGE OF REGISTER-
                         *                                    PAGE LIMITS AND/OR MEMORY LIMITS
01 001AE    351002B9               STW,1     PAGLIM
01 001AF    352002AD               STW,2     MEMIN
01 001B0    353002AB               STW,3     MEMAX
01 001B1    350002B6     REITER    STW,0     PAGE             ENTER DESIRED STARTING PAGE
01 001B2    350002B8               STW,0     PAGBEG
01 001B3    6C000010 A             RD,0      X'10'            SS2 SET
01 001B4    6840009B               BCR,4     GETPAT           REPEAT TEST WITH CURRENT PAGE LIMITS
                         *                                    AND/OR CURRENT CORE PARAMETERS
01 001B5    320002C1               LW,0      ZERO
01 001B6    350002B8               STW,0     PAGBEG
01 001B7    6800007D               B         RESET            RETURN TO RECALCULATE PARAMETERS
```

```
    592                                          PAGE
    593                               *
    594                               * ERRØR RØUTINES FØR REGISTER  PAGE TESTS
    595                               *
    596   01 001B8    320002C1       REGERR    LW,0      ZERØ                 CLEAR REGISTER ZERØ
    597   01 001B9    480002B6                 EØR,0     PAGE                 RECØRD PAGE
    598   01 001BA    48000297                 EØR,0     R                    RECØRD R
    599   01 001BB    48000288                 EØR,0     BYTCNT               RECØRD BYTE
    600   01 001BC    4800029B                 EØR,0     REGIND               REGIND=X'10000000'
    601   01 001BD    32100287                 LW,1      BYTRET               RETURN ENTRY
    602   01 001BE    351001EA                 STW,1     RETURN
    603   01 001BF    321002B9                 LW,1      PAGLIM               LØAD REG.1 WITH MAX. PAGE IMPLEMENTE
    604   01 001C0    3220028F                 LW,2      PATSTØ               LØAD REG.2 WITH CURRENT PATTERN
    605   01 001C1    3230029F                 LW,3      TEMP                 LØAD REG. 3 WITH PATTERN RECEIVED
    606   01 001C2    32400002 A               LW,4      2
    607   01 001C3    48400003 A               EØR,4     3                    LØAD REG.4 WITH DIFFERING BITS
    608   01 001C4    680001E2                 B         ERRØR
    609   01 001C5    320002C1       PAGERR    LW,0      ZERØ                 CLEAR R0
    610   01 001C6    48000293                 EØR,0     FLPCNT               RECØRD HIGHER PAGE NUMBER BITS 23-27
    611   01 001C7    48000297                 EØR,0     R                    RECØRD R IN BITS 8-11
    612   01 001C8    48000288                 EØR,0     BYTCNT               RECØRD BYTE NØ. BITS 29-31
    613   01 001C9    4800028D                 EØR,0     PGAIND               PGAIND=X'20000000'
    614   01 001CA    325002B6                 LW,5      PAGE                 RECØRD DISIRED PAGE IN R5-BITS23-27
    615   01 001CB    3210028A                 LW,1      PAGRET               RETURN ENTRY
    616   01 001CC    351001EA                 STW,1     RETURN
    617   01 001CD    321002B9                 LW,1      PAGLIM               RECØRD PAGES IMPLEMENTED IN R1
    618   01 001CE    680001E2                 B         ERRØR
```

```
    619                                          PAGE
    620                               *
    621                               * ERRØR RØUTINES FØR MEMØRY TESTS
    622                               *
    623   01 001CF    32000006 A     MACERR    LW,0      6                    FAILING ADDRESS
    624   01 001D0    480002A6                 EØR,0     MACERI               ERRØR ID
    625   01 001D1    32F002A7                 LW,F      MACRET               RETURN ENTRY
    626   01 001D2    35F001EA                 STW,F     RETURN
    627   01 001D3    321002B9       ERINFØ    LW,1      PAGLIM               MAXIMUM PAGES IMPLEMENTED
    628   01 001D4    322002AB                 LW,2      MEMAX                MAXIMUM CØRE ADDRESS
    629   01 001D5    323002A4                 LW,3      ERRCTR               ERRØR CØUNT
    630   01 001D6    324002BA                 LW,4      PASCTR               PASS CØUNT
    631   01 001D7    680001E2                 BCR,0     ERRØR
    632   01 001D8    32000008 A     MEXERR    LW,0      8                    FAILING ADDRESS
    633   01 001D9    480002A8                 AND,0     MAXCØR               ADDRESS MASK
    634   01 001DA    460002B1                 EØR,0     MEXERI               ERRØR ID
    635   01 001DB    32F002B0                 LW,F      MEXRET               RETURN ENTRY
    636   01 001DC    35F001FA                 STW,F     RETURN
    637   01 001DD    3250000C A               LW,5      C                    RESULT IN RC AFTER BIR,C
    638   01 001DE    326002B2                 LW,6      MINUS1               CØRRECT RESULT IN RC
    639   01 001DF    32700005 A               LW,7      5
    640   01 001E0    48700006 A               EØR,7     6                    DIFFERENCE
    641   01 001E1    680001D3                 BCR,0     ERINFØ
    642                               *
    643                               *
    644   01 001E2    6D000041 A     ERRØR     WD,0      X'41'                TURN ØN ALARM
    645   01 001E3    6C000010 A               RD,0      X'10'                SS4 SET
    646   01 001E4    691001E6                 BCS,1     $+2                  YES, NØ HALT ØN ERRØR
    647   01 001E5    2F000000 A     ERHALT    WAIT
    648                               * REFER TØ PAGE 4 ØF LISTING FØR AMPLIFYING ERRØR INFØRMATIØN
    649   01 001E6    6D000040 A               WD,0      X'40'                TURN ØFF ALARM
    650   01 001E7    320002A4                 LW,0      ERRCTR               INCREMENT ERRØR CØUNT
    651   01 001E8    650001E9                 BIR,0     $+1
    652   01 001E9    350002A4                 STW,0     ERRCTR
    653   01 001EA    00000000 A     RETURN    DATA      0                    RETURN BRANCH
```

```
      654
      655                                      *
      656                                      * TRAP PRØCESSING SWITCH
      657                                      *
      658                                             BØUND     8
      659    01 001EC      00000000 A    NØNØP      DATA      0
      660    01 001ED      00000000 A               DATA      0
      661    01 001EE      000001F0                 DATA      NØNØP+4
      662    01 001EF      00000000 A               DATA      0
      663    01 001F0      68000252                 BCR,0     TRAP40           INTERRUPT SYSTEM FAULT
      664    01 001F1      68000252                 BCR,0     TRAP40           MEMØRY PRØTECT VIØLATIØN
      665    01 001F2      68000252                 BCR,0     TRAP40           MØDE VIØLATIØN
      666    01 001F3      2E000000 A               WAIT
      667    01 001F4      68000162                 BCR,0     MAXMEM           NØNEXISTENT MEMØRY
      668    01 001F5      2E000000 A               WAIT
      669    01 001F6      2E000000 A               WAIT
      670    01 001F7      2E000000 A               WAIT
      671    01 001F8      68000252                 BCR,0     TRAP40           NØNEXISTENT INSTRUCTIØN
      672                                           BØUND     8
      673    01 001FA      00000000 A    UNIMP      DATA      0
      674    01 001FB      00000000 A               DATA      0
      675    01 001FC      00000254                 DATA      TRAP41
      676    01 001FD      00000000 A               DATA      0
      677    01 001FE      00000000 A    STACK      DATA      0
      678    01 001FF      00000000 A               DATA      0
      679    01 00200      00000256                 DATA      TRAP42
      680    01 00201      00000000 A               DATA      0
      681    01 00202      00000000 A    ØFLØ       DATA      0
      682    01 00203      00000000 A               DATA      0
      683    01 00204      00000258                 DATA      TRAP43
      684    01 00205      00000000 A               DATA      0
      685    01 00206      00000000 A    FLØAT      DATA      0
      686    01 00207      00000000 A               DATA      0
      687    01 00208      0000025A                 DATA      TRAP44
      688    01 00209      00000000 A               DATA      0
```

```
      689
      690                                      *
      691    01 0020A      00000000 A    DEC        DATA      0
      692    01 0020B      00000000 A               DATA      0
      693    01 0020C      0000025C                 DATA      TRAP45
      694    01 0020D      00000000 A               DATA      0
      695    01 0020E      00000000 A    TIMER      DATA      0
      696    01 0020F      00000000 A               DATA      0
      697    01 00210      0000025E                 DATA      TRAP46
      698    01 00211      00000000 A               DATA      0
      699    01 00212      00000000 A    CALL1      DATA      0
      700    01 00213      00000000 A               DATA      0
      701    01 00214      00000260                 DATA      TRAP48
      702    01 00215      00000000 A               DATA      0
      703    01 00216      00000000 A    CALL2      DATA      0
      704    01 00217      00000000 A               DATA      0
      705    01 00218      00000262                 DATA      TRAP49
      706    01 00219      00000000 A               DATA      0
      707    01 0021A      00000000 A    CALL3      DATA      0
      708    01 0021B      00000000 A               DATA      0
      709    01 0021C      00000264                 DATA      TRAP4A
      710    01 0021D      00000000 A               DATA      0
      711    01 0021E      00000000 A    CALL4      DATA      0
      712    01 0021F      00000000 A               DATA      0
      713    01 00220      00000266                 DATA      TRAP4B
      714    01 00221      00000000 A               DATA      0
      715    01 00222      00000000 A    TRAPUN     DATA      0
      716    01 00223      00000000 A               DATA      0
      717    01 00224      00000268                 DATA      TUNASS
      718    01 00225      00000000 A               DATA      0
```

```

                                        *
                                        * INTERRUPT PRØCESSING SWITCH
                                        *
                                                 BØUND    8
 01 00226     00000000 A   PØWØN         DATA     0
 01 00227     00000000 A                 DATA     0
 01 00228     0000026A                   DATA     INT50
 01 00229     00000000 A                 DATA     0
 01 0022A     00000000 A   PØWØFF        DATA     0
 01 0022B     00000000 A                 DATA     0
 01 0022C     0000026C                   DATA     INT51
 01 0022D     00000000 A                 DATA     0
 01 0022E     00000000 A   PULSE1        DATA     0
 01 0022F     00000000 A   PULSE2        DATA     0
 01 00230     00000000 A   PULSE3        DATA     0
 01 00231     00000000 A   PULSE4        DATA     0
 01 00232     00000000 A   MEMPAR        DATA     0
 01 00233     00000000 A                 DATA     0
 01 00234     0000026E                   DATA     INT56
 01 00235     00000000 A                 DATA     0
 01 00236     00000000 A   INTUN         DATA     0
 01 00237     00000000 A                 DATA     0
 01 00238     00000270                   DATA     IUNASS
 01 00239     00000000 A                 DATA     0
 01 0023A     00000000 A   CØUNT1        DATA     0
 01 0023B     00000000 A                 DATA     0
 01 0023C     00000272                   DATA     INT58
 01 0023D     00000000 A                 DATA     0
 01 0023E     00000000 A   CØUNT2        DATA     0
 01 0023F     00000000 A                 DATA     0
 01 00240     00000274                   DATA     INT59
 01 00241     00000000 A                 DATA     0
```

```

                                        *
 01 00242     00000000 A   CØUNT3        DATA     0
 01 00243     00000000 A                 DATA     0
 01 00244     00000276                   DATA     INT5A
 01 00245     00000000 A                 DATA     0
 01 00246     00000000 A   CØUNT4        DATA     0
 01 00247     00000000 A                 DATA     0
 01 00248     00000278                   DATA     INT5B
 01 00249     00000000 A                 DATA     0
 01 0024A     00000000 A   INØUT         DATA     0
 01 0024B     00000000 A                 DATA     0
 01 0024C     0000027A                   DATA     INT5C
 01 0024D     00000000 A                 DATA     0
 01 0024E     00000000 A   PCPINT        DATA     0
 01 0024F     00000000 A                 DATA     0
 01 00250     00000060     RESTRT        DATA     START
 01 00251     00000000 A                 DATA     0
```

```
                                         PAGE
                                  *
                                  * TRAP HALTS FØR IDENTIFICATIØN ØF SPURIØUS TRAPS
                                  *
                                         BØUND     8
 01 00252   2E000000 A   TRAP40   WAIT                          NØNALLØWED ØPERATIØN
 01 00253   68000060              BCR,0    START
 01 00254   2E000000 A   TRAP41   WAIT                          UNIMPLEMENTED INSTRUCTIØN
 01 00255   68000060              BCR,0    START
 01 00256   2E000000 A   TRAP42   WAIT                          PUSH-DØWN STACK LIMIT REACHED
 01 00257   68000060              BCR,0    START
 01 00258   2E000000 A   TRAP43   WAIT                          FIXED-PØINT ARITHMETIC ØVERFLØW
 01 00259   68000060              BCR,0    START
 01 0025A   2E000000 A   TRAP44   WAIT                          FLØATING-PØINT FAULT
 01 0025B   68000060              BCR,0    START
 01 0025C   2E000000 A   TRAP45   WAIT                          DECIMAL ARITHMETIC FAULT
 01 0025D   68000060              BCR,0    START
 01 0025E   2E000000 A   TRAP46   WAIT                          WATCHDØG TIMER RUN-ØUT
 01 0025F   68000060              BCR,0    START
 01 00260   2E000000 A   TRAP48   WAIT                          CALL1
 01 00261   68000060              BCR,0    START
 01 00262   2E000000 A   TRAP49   WAIT                          CALL2
 01 00263   68000060              BCR,0    START
 01 00264   2E000000 A   TRAP4A   WAIT                          CALL3
 01 00265   68000060              BCR,0    START
 01 00266   2E000000 A   TRAP4B   WAIT                          CALL4
 01 00267   68000060              BCR,0    START
 01 00268   2E000000 A   TUNASS   WAIT                          UNASSIGNED TRAP 45,47,4C-4F
 01 00269   68000060              BCR,0    START
```

```
                                         PAGE
                                  *
                                  * INTERRUPT HALTS FØR IDENTIFICATIØN ØF SPURIØUS INTERRUPTS
                                  *
                                         BØUND     8
 01 0026A   2E000000 A   INT50    WAIT                          PØWER ØN
 01 0026B   0E000250              LPSD,0   RESTRT
 01 0026C   2E000000 A   INT51    WAIT                          PØWER ØFF
 01 0026D   0E000250              LPSD,0   RESTRT
 01 0026E   2E000000 A   INT56    WAIT                          MEMØRY PARITY
 01 0026F   0E000250              LPSD,0   RESTRT
 01 00270   2E000000 A   IUNASS   WAIT                          UNASSIGNED INTERRUPT 57,5E,5F
 01 00271   0E000250              LPSD,0   RESTRT
 01 00272   2E000000 A   INT58    WAIT                          CØUNTER 1 ZERØ
 01 00273   0E000250              LPSD,0   RESTRT
 01 00274   2E000000 A   INT59    WAIT                          CØUNTER 2 ZERØ
 01 00275   0E000250              LPSD,0   RESTRT
 01 00276   2E000000 A   INT5A    WAIT                          CØUNTER 3 ZERØ
 01 00277   0E000250              LPSD,0   RESTRT
 01 00278   2E000000 A   INT5B    WAIT                          CØUNTER 4 ZERØ
 01 00279   0E000250              LPSD,0   RESTRT
 01 0027A   2E000000 A   INT5C    WAIT                          INPUT/ØUTPUT
 01 0027B   0E000250              LPSD,0   RESTRT
```

```
     822                                     PAGE
     823    01 0027C    3C3C3C3C A   PATTRN   DATA      X'3C3C3C3C'
     824    01 0027D    C3C3C3C3 A            DATA      X'C3C3C3C3'
     825    01 0027E    5A5A5A5A A            DATA      X'5A5A5A5A'
     826    01 0027F    A5A5A5A5 A            DATA      X'A5A5A5A5'
     827    01 00280    FF000000 A   BYTE0    DATA      X'FF000000'
     828    01 00281    00FF0000 A   BYTE1    DATA      X'00FF0000'
     829    01 00282    0000FF00 A   BYTE2    DATA      X'0000FF00'
     830    01 00283    000000FF A   BYTE3    DATA      X'000000FF'
     831    01 00284    00000004 A   BYTMAX   DATA      X'4'
     832    01 00285    4800029C     BYTAND   AND,0     MASK
     833    01 00286    32000280     BYTMØD   LW,0      BYTE0
     834    01 00287    680000E9     BYTRET   B         RETBYT
     835    01 00288    00000000 A   BYTCNT   DATA      0
     836    01 00289    68000060     BRST     B         START
     837    01 0028A    6800014A     PAGRET   B         RETPAG
     838    01 0028B    00000008 A   PATLIM   DATA      X'8'
     839    01 0028C    3200027C     PATSCT   LW,0      PATTRN
     840    01 0028D    20000000 A   PGAIND   DATA      X'20000000'
     841    01 0028E    00000000 A   PATCNT   DATA      0
     842    01 0028F    00000000 A   PATSTØ   DATA      0
     843    01 00290    00000000 A   CMPATR   DATA      0
     844    01 00291    000FFFFF A   FIVEFS   DATA      X'FFFFF'
     845    01 00292    00555555 A   FIVES    DATA      X'555555'
     846    01 00293    00000000 A   FLPCNT   DATA      0
     847    01 00294    32000290     LØADRG   LW,0      CMPATR
     848    01 00295    3200029D     LDMNS1   LW,0      MINS1
     849    01 00296    3200028F     LØDPAT   LW,0      PATSTØ
     850    01 00297    00000000 A   R        DATA      0
     851    01 00298    00000000 A   RR       DATA      0
     852    01 00299    00800000 A   R8       DATA      X'800000'
     853    01 0029A    01000000 A   RMAX     DATA      X'1000000'
     854    01 0029B    10000000 A   REGIND   DATA      X'10000000'
     855    01 0029C    00000000 A   MASK     DATA      0
     856    01 0029D    FFFFFFFF A   MINS1    DATA      X'FFFFFFFF'
     857    01 0029E    3500029F     STRPAT   STW,0     TEMP
     858    01 0029F    00000000 A   TEMP     DATA      0
```

```
     859                                     PAGE
     860    01 002A0    00000810 A   ARMINT   DATA      X'810'
     861    01 002A1    65C00197     BIRLØD   BIR,0     EXRET
     862    01 002A2    68000000 A   BTØBIR   BCR,0     0
     863    01 002A3    680001D8     BTØERR   BCR,0     MEXERR
     864    01 002A4    00000000 A   ERRCTR   DATA      0
     865    01 002A5    2E000000 A   HALT     WAIT
     866    01 002A6    30000000 A   MACERI   DATA      X'30000000'
     867    01 002A7    6800017F     MACRET   BCR,0     MACRET
     868    01 002A8    0001FFFF A   MAXCØR   DATA      X'0001FFFF'
     869    01 002A9    0000020F A   MAXPAG   DATA      X'20F'
     870    01 002AA    7FFFFFFF A   MAXPØS   DATA      X'7FFFFFFF'
     871    01 002AB    00000000 A   MEMAX    DATA      0
     872    01 002AC    35100300 A   MEMFIL   STW,1     X'300'
     873    01 002AD    00000000 A   MEMIN    DATA      0
     874    01 002AE    32500000 A   MEMLØD   LW,5      0
     875    01 002AF    35600000 A   MEMSTØ   STW,6     0
     876    01 002B0    68000197     MEXRET   BCR,0     EXRET
     877    01 002B1    40000000 A   MEXERI   DATA      X'40000000'
     878    01 002B2    80000000 A   MINUS1   DATA      X'80000000'
     879    01 002B3    FFFE0300 A   N300     DATA      X'FFFE0300'
     880    01 002B4    00000001 A   ØNE      DATA      1
     881    01 002B5    00000300 A   P300     DATA      X'300'
     882    01 002B6    00000000 A   PAGE     DATA      0
     883    01 002B7    00000000 A   PAGEP    DATA      0
     884    01 002B8    00000000 A   PAGBEG   DATA      0
     885    01 002B9    00000000 A   PAGLIM   DATA      0
     886    01 002BA    00000000 A   PASCTR   DATA      0
     887    01 002BB    00000000 A   PASTAG   DATA      0
     888    01 002BC    0000000F A   REGLIM   DATA      F
     889    01 002BD    00000000 A   SAVERR   DATA      0
     890    01 002BE    35B00000 A   STØBCR   STW,B     0
     891    01 002BF    35A00000 A   STØBIR   STW,A     0
     892    01 002C0    35900000 A   STØRE9   STW,9     0
     893    01 002C1    00000000 A   ZERØ     DATA      0
     894    01 002C2    FFFFFFF0 A   7FSZRØ   DATA      X'FFFFFFF0'
     895           01 00060                   END       START
```